PRINCIPLES FOR TESTING A DATA FUSION SYSTEM

March 31, 1998

R. N. DeWitt
Pacific-Sierra Research Corporation
1400 Key Boulevard, Suite 700
Arlington, Virginia 22209

ABSTRACT

The testing of a data fusing system presents a conceptual challenge. What seem to be simple and direct approaches to testing may be tantamount to making a judgment on the basis of the flip of a coin. In this paper we dissect the intelligence system and identify precisely the role of the fusion engine in it, with particular emphasis on distinguishing between the deterministic parts of the system and those that are sources of noise or randomness. It is emphasized that the role of the fusion engine is to perform a deterministic calculation of a probability distribution through a combining of more elementary probability distributions characterizing the sources of noise or randomness in the overall system.

While such a deterministic calculator can be verified by comparing the results of the fusion engine's calculation to results that are computed by some independent means, it is recognized that many users would be more comfortable with a test in which the response of the fusion engine is compared in some fashion to the ground truth (or simulation thereof) that gave rise to the response. Toward this end we present a valid and practicable testing procedure which results in a chi-square comparison of the output distribution of the fusion engine with an appropriately derived histogram of instantiations of ground truth (or simulations thereof). It is shown that this test would be equivalent to estimating independently the desired output of the fusion engine by means of a Monte Carlo calculation. Finally, some other considerations and judgment criteria applicable to the fusion engine are mentioned.

1. INTRODUCTION AND PROBLEM STATEMENT

After a number of opportunities to criticize test plans developed by others for the fusion engine that we designed, which we call simply the Tracker, it was only during the past year that an opportunity arose to put down our own thoughts on how such testing should be done. Some aspects of my own considerations differ from past ideas and suggest a test that differs somewhat from those that had been proposed earlier; perhaps they will be of interest to this audience.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (FROM - TO)
31-03-1998	Conference Proceedings	xx-xx-1998 to xx-xx-1998

4. TITLE AND SUBTITLE
Principles for Testing a Data Fusion System
Unclassified

5a. CONTRACT NUMBER
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
DeWitt, R. N. ;

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME AND ADDRESS
Pacific-Sierra Research Corporation
1400 Key Boulevard, Suite 700
Arlington, VA22209

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME AND ADDRESS
Director, CECOM RDEC
Night Vision and Electronic Sensors Directorate, Security Team
10221 Burbeck Road
Ft. Belvoir, VA22060-5806

10. SPONSOR/MONITOR'S ACRONYM(S)
11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
APUBLIC RELEASE
,

13. SUPPLEMENTARY NOTES
See Also ADM201041, 1998 IRIS Proceedings on CD-ROM.

14. ABSTRACT
The testing of a data fusing system presents a conceptual challenge. What seem to be simple and direct approaches to testing may be tantamount to making a judgment on the basis of the flip of a coin. In this paper we dissect the intelligence system and identify precisely the role of the fusion engine in it, with particular emphasis on distinguishing between the deterministic parts of the system and those that are sources of noise or randomness. It is emphasized that the role of the fusion engine is to perform a deterministic calculation of a probability distribution through a combining of more elementary probability distributions characterizing the sources of noise or randomness in the overall system. While such a deterministic calculator can be verified by comparing the results of the fusion engine?s calculation to results that are computed by some independent means, it is recognized that many users would be more comfortable with a test in which the response of the fusion engine is compared in some fashion to the ground truth (or simulation thereof) that gave rise to the response. Toward this end we present a valid and practicable testing procedure which results in a chi-square comparison of the output distribution of the fusion engine with an appropriately derived histogram of instantiations of ground truth (or simulations thereof). It is shown that this test would be equivalent to estimating independently the desired output of the fusion engine by means of a Monte Carlo calculation. Finally, some other considerations and judgment criteria applicable to the fusion engine are mentioned.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE	Public Release	13	Fenster, Lynn lfenster@dtic.mil
Unclassified	Unclassified	Unclassified			19b. TELEPHONE NUMBER International Area Code Area Code Telephone Number 703767-9007 DSN 427-9007

From the often heated discussions that arose when I was in the role of a critic of test plans, as well as during the current considerations, it was apparent that views on testing differ from one person to another, and these views are sometimes quite strongly held. Hoping to establish a noncontroversial starting point, I would like to propose a principle borrowed from the medical profession: "First, do no harm." Translated to our problem of testing fusion engines, this becomes the following*: A system test should be designed so that it will be passed with certainty if the system performs exactly as intended.* Of course this leaves unquestioned whether the system designer's intent was valid, but let's put the design question aside for the moment and consider the test to apply only to the implementation of the system. The point here is that a test that cannot be relied upon regarding the simple question of correct implementation of the system can no more be relied upon to give a correct outcome in the more general question of system validity.

But would anyone really propose a test that a correctly working system might fail? Indeed they would. Typically the test is proposed in the form of "Let's feed in observational reports of a situation where the ground truth is known and see if the fusion engine assigns the *highest probability* to that state that is known to be true." Now it may be that the combination of observations leads to a completely unambiguous inference, in which case a correctly designed and implemented system should be expected to pass this test with certainty, but if such cases were the norm, there would hardly be a need for a probabilistic basis for the fusion system. More often the observational reports, even after being correctly fused and correlated, will result in probability being distributed over a considerable number of states or hypotheses, and whether that state that corresponds to ground truth has been assigned the highest probability depends on the luck of the draw, for example, on the specific realization of noise processes that occurred in the various sensors at the time when they made their observations. Such a test, if used to reject the system under test, would risk violating the principle "First, do no harm," but rarely do things get to that harmful stage before sanity is restored, and the question becomes one of "Does the system assign any probability at all to the state corresponding to ground truth?"

Reduced to this form the test has degenerated to a rather weak form of a Popperian hypothesis test [Popper, 1959], and the fusion system under test would be regarded to pass unless it assigns no probability at all to the state that corresponds to the ground truth, that is, unless the fusion system somehow infers that the state known to be true is impossible in the light of the available observations and background information. Yet even in the unlikely event that the fusion system were to fail this weakened test, its rejection might well be unjust and a violation of the dictum "First, do no harm." This possible exculpation results from regarding the fusion system as a research program in the manner of Lakatos [1978] rather than a theory or hypothesis to be tested in the manner of Popper, and this will be discussed in a later section.

Much of the conceptual difficulty regarding tests of fusion systems derives from the probabilistic nature of their product. But this characterization of the fusion system's product is itself ambiguous, and so the following section considers what is meant by a probabilistic system and the extent to which the phrase applies to a fusion engine and its product.

2. NOISE SOURCES, NOISE MODELS, AND PROBABILITY CALCULATIONS

The term *probabilistic system* seems to be applied to fusion engines such as Tracker because they are very much concerned with probability in the sense that their purpose is to compute probability distributions. However, for many people the same term may evoke the idea of a system that behaves in a probabilistic fashion, that is, one whose output does not predictably correspond on a one-to-one basis to its input. Such a system can be modeled as including a source of noise which leads to an unpredictable output. A probabilistic system in this sense is quite different from a fusion engine such as Tracker; we expect

Tracker to be quite deterministic in the sense that if a given stream of reports is entered into Tracker repeatedly we expect the same result every time the stream is entered. In order to avoid further confusion on this point let us adopt two different terms. It would be better to refer to Tracker as a *probability calculator*, while the systems that are not deterministic in their behavior and that must be modeled as including a noise source can be referred to as *probabilistic phenomena*, and it is this latter meaning that we might assume to be intended by anyone using the term *probabilistic system*.

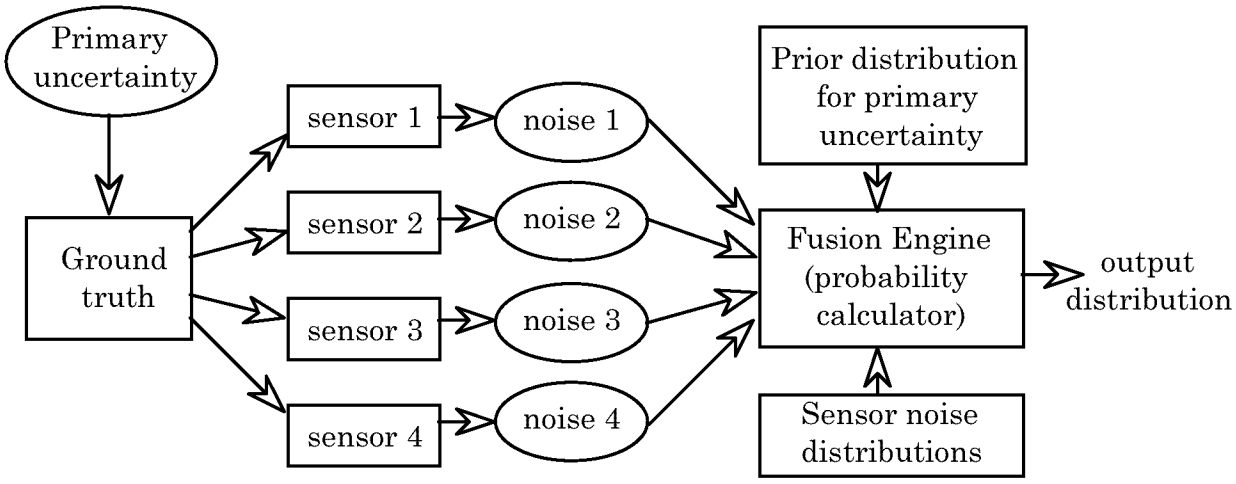

Figure 1. Intelligence system dissected to isolate noise phenomena from probability calculator.

Clearly any system that includes a probabilistic phenomenon itself becomes a probabilistic phenomenon. Thus, we can characterize Tracker as being deterministic only by separating it from the suite of noisy sensors from which it receives its input stream. If we consider the intelligence system, comprising Tracker, the sensors, and the various models, we are back to a probabilistic phenomenon or probabilistic system.

Many of the differences that we have had in discussing Tracker testing arise from a desire to test Tracker as though it were the entire intelligence system, including the sensor noise sources. End-to-end tests of the intelligence system are, of course, desirable, but such testing calls for a large commitment of resources, inasmuch as it would require conducting exercises in which reports are generated by observing real objects on the ground with the real sensors. Those arguing that such comprehensive testing can be applied to Tracker often suggest that the testing can be carried out within available resources by simulating with a computer the ground truth and the sensors' response to the ground truth. A closer look at that assumption is part of the discussion to follow.

A deterministic probability calculator such as Tracker is expected to compute a probability distribution by combining a number of models (conditional probability distributions and prior probability distributions). If we look upon these models as just another set of inputs to Tracker, the action of Tracker is reduced to the simple mechanical processes of selecting particular models appropriate to a newly received message and combining them with the prior distribution to effect a new updated probability distribution. The end result is to determine a probability distribution that in some sense is better suited to the needs of the user than would be the more elementary distributions that have gone into calculating it. Usually the resulting distribution is over a set of hypothetical responses to a query posed by the user, while the input distributions relate to the noise processes within the sensors, as well as to the intrinsically uncertain behavior of the observed object for which state estimates or predictions are sought in the user's queries.

3. TESTING AND CHARACTERIZING A PROBABILITY CALCULATOR

Because the action of the probability calculator is determined by an algorithm, testing it in isolation should be a rather straightforward process of assuring that the calculation proceeds as intended. The testing should answer the questions (1) does the calculator do what it is expected to do on the basis of a single message, (2) does it do the calculation consistently if the Tracker is repeatedly prepared to the same initial state and subjected to the same message, and (3) does the Tracker correctly handle successive messages so that it can respond to a message stream? A test of question 3 might include a test for commutability in which two messages are processed in different orders. Provided that the times between the two corresponding observations are so short as to not allow a state transition of the object of interest, the resulting state of Tracker should be the same with either ordering.

Tests of the kind described above would serve to establish that a probability calculator such as Tracker is performing correctly the deterministic function that is expected of it. Another set of tests can be considered for the purpose of characterizing the probability calculator. Such a characterization is desirable because a probability calculator such as Tracker is a complex enough system that when subjected to an extended stream of messages it is difficult to anticipate exactly what state Tracker will assume and how quickly it will assume it. In particular, we cannot be sure what will be the effect of small changes in the input stream. The following subsections deal with some of the possible characters that such testing might indicate.

3.1. DETERMINACY

A system module is regarded as deterministic if a particular initialization and input data stream always results in the same output. This is equivalent to saying that the module includes no noise source. Because a probability calculator is by definition expected to be deterministic, a test for consistency is an obvious basic test for Tracker, and so has already been mentioned above in the context of the tests made to assure that Tracker is correctly carrying out its algorithm. The tests for determinacy mentioned above applied to the processing of only a few messages. A more thorough test might be to assure that the same final Tracker state is always achieved after the processing of a prolonged stream of messages.

3.2. RESPONSIVENESS

We expect a change of the input stream to Tracker to result in a change in its output. A system would be deterministic if it always produced the same output, irrespective of the input. But such a system would not be responsive. A failure to be responsive can come about in various ways. As quantization is used to reduce the number of states, and thus the information content, with which Tracker must deal, there is the possibility that certain changes of the input may be too small to effect a change of state at the granularity that is being used. Because granularity will be chosen to provide an output that is satisfactory to the user, the concern here is not simply the lack of response to a small change of input; we are not concerned about a failure to respond to a variation at the input that would lead to a variation of the output that is smaller than the acceptable quantization level. The real concern is that there may arise quantization effects by which significant variations of the output are blocked by quantization employed at some intermediate stage of the processing.

A second kind of failure of responsiveness can occur from saturation effects. A simple example of this occurs when a meter of some kind is subjected to an input that exceeds its range, and the indicating needle is "pegged". Again there will be cases where such effects are introduced intentionally by building the saturation effect into one or the other of the models, thereby implementing some form of the law of diminishing returns. It is the unintended occurrences of saturation that are of concern. A noteworthy

example of this kind of problem may have been seen when Tracker failed to respond to a report (i.e., to regard it to be a valid report) because the hypothesis management pruned away world views, resulting in a situation in which all surviving world views accounted for as many objects as were believed to exist, with the result that no new objects could be hypothesized. We have discussed other aspects of the danger of hypothesis pruning in an earlier paper. [DeWitt, 1997]

3.3. CHAOTIC OR CATASTROPHIC BEHAVIOR

If a small change of the input stream results in a large change in the output, the system is showing either chaotic or catastrophic behavior. Either of these behaviors can be consistent with the system being deterministic. Hypothesis pruning is again a possible spurious source of this character in Tracker; a small change in the input can cause the retention of hypothesis A at the expense of hypothesis B and the difference between the two may be large.

The term catastrophic seems appropriate to an immediate large change in the output because of a small change in the input. The term chaotic is more general and would apply even if the large change evolved only gradually over the course of an extended observation stream.

Testing for these characterizations would involve establishing a repeatable initialization and subjecting Tracker to pairs of message streams that slightly differ in some manner. A measurement of the Bhattacharyya similarity of the ultimate tracker states would allow the presence of chaotic behavior to be discerned [Bhattacharyya, 1943].

Given that chaotic behavior occurs, the question arises as to whether this is an indication of a problem. One can conceive of such behavior arising as a form of error that may be introduced by the various cellularizations and quantizations introduced to limit the size of the calculation, but one can also conceive of critical observations that distinguish two radically different outcomes. Thus, chaotic behavior should be looked upon as a reason for concern regarding, but not necessarily rejection of, the fusion engine.

3.4. ATTRACTORS

The discussion above treats chaotic behavior as though it were a possibility to be expected of Tracker. Perhaps a better way to think of it is as a possible behavior that may be expected with certain movement models. The movement models that we have worked with for the most part would not be expected to result in chaotic behavior. If the state of the system were projected to a time long after the last observation of the test stream, the system would gradually assume the state representing maximal entropy, in which all state probability distributions (SPDs) revert to their corresponding latent SPDs. This ultimate state can be referred to as an attractor in the dynamics of Tracker.

However, it is possible to conceive of movement models that when used in Tracker would lead to a phase portrait in which there would be a more elaborate attractor. This could happen, for example, if the state space for the objects were to partition into two or more subsets between which there would be no possibility of transition. For example, were the movement of the objects of interest restricted to roads and if the road network disconnected into two or more unconnected lesser networks, the ultimate projection of the SPD would assign probability to the "sublatents" in some proportion that would be determined by the distribution of probability at the time at which the network disconnection was assumed to have occurred. In such a case the division of probability among the basins for the extended attractors can be affected by new observational reports, but not by projection.

4. FALSIFICATION OF A PROBABILISTIC PREDICTOR

A fusion engine such as Tracker is driven by a message stream produced by the interaction of the ground truth with one or more sensors and their subsequent processing. In part because the contingencies of the sensing process are too many to be quantified and in part because of the intrinsically stochastic nature of the detection/measurement process, the message stream must be modeled as a probabilistic phenomenon. The uncertainty regarding the ground truth that has led to any particular message is dealt with by means of a probability distribution that we refer to as the observation model. The model is defined in terms of the distribution of possible messages that may be produced from the sensor and its associated processing when subjected to a specific ground truth. In Tracker's updating process this conditional probability is then inverted by applying Bayes' rule in conjunction with some prior distribution over states (hypothesized ground truth) that may have been shaped in part by earlier messages.

The relationship that the output from Tracker bears to the ground truth is of great interest, but this relationship is characteristic of the entire intelligence system, not of Tracker in isolation. Ideally, one would like to have the output of an intelligence system identify with certainty a single hypothesized state for the object of interest, but, because sensing is a probabilistic phenomenon, the relationship between ground truth and the sensor's message cannot be assumed to be consistent from one instance to the next. The fusion engine (Tracker) must make allowance for the uncertainty of the relationship, which it does by means of the observation model. If the message stream reports many observations of an object that can be expected to remain static over the period of the observations, the uncertainty may be reduced by compounding the observational reports, perhaps to the point at which the object can be estimated to be almost certainly in some specific state. The unambiguous character of such a case should be regarded as characteristic of the message stream, not of the fusion engine or Tracker.

In most cases the message stream will not be unambiguous, and it is the responsibility of Tracker to call attention to the various alternative interpretations (hypotheses) that can be derived from the message stream, which it does by outputting a probability distribution that may extend over many possible states. In such a case, one may ask, just what constitutes an incorrect response on the part of the overall intelligence system to a given ground truth? This is not a trivial question; it is by no means obvious how one should go about verifying a probabilistic predictor such as the intelligence system is intended to be. If the intelligence system has assigned the least amount of probability to the state representing the ground truth, the response is logically justified. The one logical error that it can make would occur if the intelligence system were to assign zero probability to what is the true state. Discovering such an instance would constitute falsification of the system in the Popperian sense, and, if one were to follow strictly the Popperian criterion, this would constitute a disproof of the validity of the system.

While the epistemology of Popper [1959] has enjoyed great popularity and has been very influential to the practice of science, it does not represent the most sophisticated thought on the way science is and should be practiced. Duhem has established that many scientific theories have been regarded not to have been falsified by conflicting observational statements, even though the falsifying observational statements could be regarded as infallibly true. I. Lakatos [1978] has identified one reason for this anomaly in terms of his paradigm of the "methodology of scientific research programmes". A summary of these contending lines of thought can be found in Howson and Urbach [1989]. According to Lakatos, a research program takes the form of a central "hard core" theory (in our case, that of Bayesian inference), which is combined with a "protective belt" of auxiliary assumptions or hypotheses in order to apply the theory to a particular situation and to make predictions. In our case these auxiliary assumptions or hypotheses would be the various models regarding the utilization of terrain, the probabilistic relationship between state of the object and the observational report, and the probability of the transitioning of objects of interest from one state to another. The auxiliary assumptions are "protective" in the sense that it is they, rather than the central theory, that are revised when a prediction is shown to be false. A quite convincing example of this, cited by Lakatos, occurred when astronomers, rather than rejecting Newton's laws of motion and

gravitation upon observing deviations from Newtonian predictions of planetary and cometary orbits, instead reconsidered the contemporary assumptions regarding the number and locations of the planets, and by so doing were rewarded by the discovery of new planets.

The applicability of these ideas to Tracker can be seen if we consider how we would deal with a full scale test in which it was found that the intelligence system had predicted zero probability for a site at which the object of interest truly sited in a military exercise or in some other ground truth situation. Given that Tracker has been thoroughly tested with regard to its correct handling and calculation of the models that are provided to it, the prediction of zero probability can have come about in only a few ways: (1) the terrain for the occupied site may have been rated as unsatisfactory by the terrain rating model, (2) the occupied site may lie at a distance from a prior known location for the object beyond what is deemed the maximal relocation distance for the time available for the move, or (3) the true location may lie outside the envelope of what is regarded as possible errors relative to the location reported by the sensor.

If such errors seem to occur with a frequency that exceeds what can be expected for the tail of the relevant probability distribution, a corrective action is immediately available to us; we need only modify the relevant model (i.e., the auxiliary hypothesis in the manner described by Lakatos). However, one should not be too ready to make ad hoc changes to the models solely on such a basis. Insofar as the models derive from prior knowledge and insights gained from the experience of the modeler, any such adjustment of the model should be undertaken only to the extent that it can be rationalized or independently verified. For example, an effort should be made to determine whether a parcel of terrain has been mischaracterized in the data base before it is decided that some class of terrain previously judged unsatisfactory is now to be regarded as satisfactory; similar caution should be exercised to avoid succumbing to unrestrained ad hocism in revising any other of the models. After all, the astronomers did not simply adopt the assumption that there was another planet out there perturbing the orbit that they had been studying; they went to their telescopes to verify their revised hypothesis.

5. TESTING BEYOND FALSIFICATION

Falsification tests of the kind discussed in the previous section seem to have been developed with deterministic theories in mind, rather than probabilistic ones. Their application to a probabilistic theory, such as quantum mechanics or optics, would amount to verifying only that the predicted nulls in the probability distribution matched the locations of nulls observed in experiments. Certainly survival of tests of this kind would be necessary for the theory to be credible, but we really expect more of a probability predictor; we expect that the nonzero probabilities that are assigned to states outside the null region be valid in some sense. This kind of validation cannot be accomplished with a single trial; any single outcome of one trial that lies outside the set of predicted null states is a logically possible outcome in the Popperian sense, but such an outcome cannot "validate" the prediction, it has merely failed to falsify the prediction. The validation of the kind we seek of a predicted distribution can only be accomplished by repeatedly conducting trials, thereby accumulating a sample or population whose histogram can be compared to the predicted distribution. In accumulating the sample we might attempt repeatedly to prepare the system in some manner consistent with the assumptions used to make the predictions, and to observe the outcome, grouping them into a set of predicted classes. Then the number of outcomes in each class can be compared to the predictions for that class by some such test as the chi-square test. At this point we have reduced the problem to one for which there is a considerable body of literature. Not even the literature in this area is free from epistemological controversy, but it is not our intent to attempt to plow any new ground here. Instead, we wish only to look more closely at the question of how we can go about accumulating the sample or population of trials that would be satisfactory for a validation of this kind.

5.1. CONSIDERATIONS OF TESTING BY INSTANTIATED GROUND TRUTH

Tracker computes a probability distribution over a set of hypothetical states, a typical one of which we here denote h_i. The probability distribution is determined by the report r_j that is evoked from the sensor subsystem by a ground truth t, but the relationship between the ground truth and the report is random, characterized by the conditional probability distribution $P(r_j/t)$, while the relationship between the report and the probability that Tracker computes for the hypothesis h_i is deterministic, and this relationship can be denoted $H_{i,j}$. At this point it seems advisable to introduce a change in terms, lest we confuse ourselves talking about probabilities that the intelligence system estimates, which are themselves random numbers that are distributed according to a probability distribution that we wish to discuss. Let us refer to as random estimates the numbers $H_{i,j,}$ that are instantiated when the ground truth t evokes the random report r_j. Note that t is not included in the notation because once we are given that the report j has been instantiated, t has no effect on our expectations regarding $H_{i,j,}$. Also note that if we need to we will be able to appeal to the probability-like properties of the $H_{i,j}$, that is,

$$0 \leq H_{i,j} \leq 1$$

and

$$\sum_i H_{i,j} = 1 \quad \text{for all } j.$$

The question we have to address is how we can go about generating a sample of ground truths, a typical one of which we denote t_k such that a histogram of their distribution can be compared to an output of the Tracker, for example by means of a chi-squared test. At the outset, to keep things simple, let us suppose that we want to test the response to only a single observation. The results of these considerations will then provide guidance for extending the test to more elaborate scenarios, if such elaboration seems useful and practicable. One question that we must resolve is how to distribute the ground truth instantiations that the sample is to comprise. There would seem to be three possibilities to consider, only one of which, of course, would be correct. We can (1) choose a single ground truth and keep it the same for all trials in the sample, or (2) generate a sample of ground truths that are distributed according to the prior or latent SPD, or (3) generate a sample of ground truths that are distributed in the manner implied by a some particular observational report, when used to update the prior or latent distribution. Each ground truth instantiation will be observed with the sensor, which will produce a message or report r_j from the sensor system to the Tracker. These reports are random functions of the ground truth t_k , being affected by the noise process intrinsic to the particular sensor. This random process is characterized by a distribution $P(r_j|t_k)$, which we refer to as the observation model, a known function that resides in the Tracker where it is used to interpret the reports that are received.

Because we are talking about a set of discrete reports and a set of discrete locations (cells), we can reduce the discussion to one of generating a pair of appropriately distributed random integers, (j, k). Because we are now discussing a test of the entire intelligence system, it would be necessary to generate the random pairs by actual siting of equipment at the corresponding locations and observing and reporting observations of the equipment with real sensors. The subsections below further discuss and criticize each of the three ways in which k can be distributed.

5.1.1. Test with a fixed ground truth

If the ground truth were kept the same for each trial of the sample the randomness of the random estimates $H_{i,j}$ would stem solely from the process for generating the reports, and these are estimated to be distributed according to the conditional distribution $P(r_j|t_k) = P(j/k)$. Each choice of a j determines which of the deterministic functions $H_{i,j}$ will be used to estimate the distribution of belief among the output

hypotheses h_i used by Tracker to report its results. Presumably the goal of the test will be to compare some form of sample average of the output of Tracker to the input distribution. This sample average of the output can be denoted:

$$\langle H_{i,j} \rangle_{\text{sample}} = \langle H_{i,j} | k = \kappa, \text{ a fixed value} \rangle = \frac{1}{N} \sum_{\text{trials in sample}} H_{i,j|\kappa}$$

The expected values for the sample means for each i can be written in the Lesbesque form

$$\langle H_{i,j} \rangle_{\text{expected}} = \sum_{j=1}^{J} H_{i,j} P(j \mid \kappa)$$

Note that the result is an estimated distribution that is formed by averaging a randomly selected set (sample) of deterministic functions $H_{i,j}$, each of which has the character of a probability distribution, in particular, always being positive over its nonzero range. The resulting averaged distribution cannot be expected to be comparable to the single ground truth that was used to select the random sample of reports. The output of Tracker would have the character of a distribution (albeit fuzzed out by averaging) which would have to be compared with a single discrete "true value." The comparison would necessarily be poor unless each and every one of the functions that receives significant weight in the Lesbesque form above happens to concentrate belief at the same Tracker output hypothesis $i=k$. Now such a thing could conceivably happen if the ground truth were located on the only suitable terrain for many miles around, but no one would accept such a contrived situation as a valid test of the intelligence system or of Tracker. Aside from this special case, to hold Tracker to such a standard in the face of the uncertainty introduced by the sensor system would be in effect to ask that Tracker act as a dowsing rod, that is, as a source of occult information that is able to overcome the ambiguity intrinsic to the available information.

5.1.2. Latent distribution of ground truth

The second possibility would be to distribute the instantiations of ground truth in the sample according to what we refer to as the latent distribution, which is our best estimate of how the objects of interest would be distributed based only on that evidence available to us prior to any observations. Alternatively, we could distribute the ground truth instantiations according to some other state probability distribution (SPD), so long as that SPD were available to Tracker to use as a prior distribution. If we denote this prior distribution $P(t_k) = P(k)$, we can regard each trial within the sample as represented by three integers (i, j, k), where i indexes a hypothesized state, j indexes the message that determines the distribution estimated by Tracker, and k indexes the true state of the observed object that leads to the message j from the sensor. The sample distribution of these random triplets can be written

$$P(i,j,k) = P(i \mid j,k) P(j \mid k) P(k) = H_{i,j} P(j \mid k) P(k)$$

The sample mean for the belief assigned to hypothesis i in this case would be

$$\langle H_i \rangle_{\text{sample}} = \frac{1}{N} \sum_{\substack{\kappa \text{ distributed as } P(k) \\ j \text{ distributed as } P(j|\kappa)}} H_{i,j,\kappa}$$

where N is the population of the sample. The expected value of the sample mean would be

$$\langle H_i \rangle_{\text{expected}} = \sum_{k=1}^{K} \sum_{j=1}^{J} P(i,j,k) = \sum_{k=1}^{K} \sum_{j=1}^{J} H_{i,j} P(j \mid k) P(k).$$

This distribution would be compared with the distribution of the ground truth or to a histogram of the ground truth realizations. A simple sample average would result in the random estimates outputted by Tracker being distributed all over the map, to be compared with the ground truth histogram which would be similarly distributed all over the map, a not very satisfactory basis for comparison. The two distributions would be similar in their diffuseness, but the comparison would have no particular relevance to the utility of the intelligence system.

A more relevant variation on this approach would be to compare the distribution of belief that Tracker outputs when it receives some particular message to the distribution of those instantiations of ground truth that have evoked the instantiations of that message from the sensor. Here we would expect a valid comparison that can be evaluated by such tests as chi-squared, and the test corresponds to the real-world situations in which particular messages are instantiated. A procedure to accomplish this would be to generate ground truth instantiations that are distributed in the manner described above, but to sort out the trials according to the random message r_j that is generated for each ground truth. Then the distribution of the ground truths k that gave rise to message $j=l$ can be compared to the deterministic function $H_{i,l}$ for each $i=k$.

Because $H_{i,j}$, the output estimate of belief that is produced by Tracker, is deterministic, it need only be computed once for each message r_l. Only the sample of ground truth instantiations need be generated and organized into distributions according to the random message that is instantiated in the same trial. Thus, the frequency f_i at which the ground truth $k=i$ occurs in association with the message r_l would be given by

$$\langle f_i \rangle_{\substack{j=\lambda \\ \text{sample}}} = \frac{\sum_{\substack{k \text{ distributed as } P(k) \\ j \text{ distributed as } P(j|k)}} \delta_{j\lambda}\delta_{ki}}{\sum_{\substack{k \text{ distributed as } P(k) \\ j \text{ distributed as } P(j|k)}} \delta_{j\lambda}}.$$

The expected value of the frequency for $k=i$ and $j=l$ is given by

$$\langle f_i \rangle_{\substack{j=\lambda \\ \text{expected}}} = \frac{\sum_{k=1}^{K} P(i,\lambda,k)}{\sum_{i=1}^{I}\sum_{k=1}^{K} P(i,\lambda,k)} = \frac{\sum_{k=1}^{K} H_{i,\lambda} P(\lambda \mid k) P(k)}{P(\lambda)} = \frac{H_{i,\lambda}}{P(\lambda)} \sum_{k=1}^{K} P(\lambda \mid k) P(k) = H_{i,\lambda}$$

which indicates that the two distributions are expected to match, and therefore can be compared using such tests as chi-squared.

5.1.3. Ground truth distributed as $H_{i,l}$

The final one of the distributions we considered for the test sample of ground truth instantiations is the distribution that matches the distribution of belief among hypothetical locations that Tracker assigns in response to a given message from the sensor system. While this sounds a bit like the distribution that was discussed in the preceding section, there is a significant distinction. In the preceding section the instantiations of ground truth were distributed according to the latent distribution (or to a known prior), and of these the subset that led to the instantiation of a selected message were selected and represented in a histogram for comparison with the belief distribution that Tracker calculates in response to that message. Here we are considering distributing the ground truth at the outset in accordance with the calculated belief distribution that results from a particular message, but there is no assurance that the message evoked from any ground truth instantiation will be the particular message under consideration.

Thus, the sample average over the Tracker outputs that are elicited from the sample of ground truths will include responses from a variety of messages. We may write for the sample average

$$\left\langle H_i \right\rangle_{\substack{\lambda \text{ distribution of } k \\ \text{sample}}} = \frac{1}{N} \sum_{\substack{k \text{ distributed as } H_{k,\lambda} \\ j \text{ distributed as } P(j|k)}} H_{i,j}$$

and for the expected value of this sample average

$$\left\langle H_i \right\rangle_{\substack{\lambda \text{ distribution of } k \\ \text{expected}}} = \sum_{k=1}^{K} \sum_{j=1}^{J} P(i,j,k)$$

$$\left\langle H_i \right\rangle_{\substack{\lambda \text{ distribution of } k \\ \text{expected}}} = \sum_{k=1}^{K} \sum_{j=1}^{J} P(i \mid j,k) P(j \mid k) P(k),$$

$$\left\langle H_i \right\rangle_{\substack{\lambda \text{ distribution of } k \\ \text{expected}}} = \sum_{k=1}^{K} \sum_{j=1}^{J} H_{i,j} P(j \mid k) H_{k,\lambda}$$

so the distribution of this sample-averaged belief can be expected to be rather more diffusely distributed than the sample of ground truth instantiations that gave rise to it; a chi-square comparison of the two distributions would be preordained to fail.

5.2. SUMMARY

In the three subsections above we have considered how a set of test instantiations of ground truth should be distributed in order to provide a valid comparison with Tracker's output belief distributions using such measures of correspondence of distributions as chi-squared. We examined three possible distributions, and for one of them we considered two modes of analysis.

In the first method the instantiation of ground truth is held fixed for all trials in the sample. Because the sensor system produces random messages as a result of its imperfections, and because this randomness is taken into account in inferences that Tracker draws from the messages, the Tracker output distributes belief over several hypothetical locations or states. This distribution of belief by Tracker might be a correct response to the circumstances, and the test would be appropriate to characterize the overall intelligence system or, if the message stream is analyzed, the sensor system in isolation, but it is not appropriate for testing the fusion engine or Tracker in isolation.

In the second method instantiations of ground truth are distributed according to the latent distribution for the object of interest. The resulting distribution of ground truth, if treated as a single sample, is very diffuse and so correspondingly is expected to be the output of Tracker, if averaged over all trials. These diffuse distributions are not at all representative of any operational situation, and the comparison made in such a way, even though it might be quite good, would provide little basis for judging the performance of the intelligence system. A rather better result is obtained if the instantiations of ground truths, again distributed according to the latent distribution, are paired with corresponding instantiations of the sensor response (in the form of a random-noise-influenced message). If the instantiation pairs are sorted into subsets on the basis of the message, the distribution of ground truths for each subset can be expected to match the distribution of belief that is evoked from Tracker by the corresponding message. Comparisons of this kind would constitute a valid check of Tracker. The comparison might be looked upon as a comparison of Tracker's calculated result with a Monte Carlo estimate of the same calculation.

In the third method that was considered, the sample of ground truth instantiations is distributed according to the belief distribution that Tracker outputs in response to some particular message from the sensor system. The result is a moderately diffuse distribution of ground truth instantiations that is to be

compared to a sample average of Tracker output distributions that can be expected to be rather more diffuse. A comparison of this kind is not valid; failure is built into the test procedure. There is no correspondence of such a test with any kind of operational situation, and nothing to commend it over the first test as a means for characterizing the entire intelligence system or the sensor suite.

5.3. SIMULATION VERSUS GROUND TRUTH

Above we have identified the procedure that can be applied to test the entire intelligence system; the distribution of samples of ground truth instantiations that give rise to a particular message from the sensor are compared to the distribution computed in the Tracker in response to that message. The test assumes that real objects of interest are to be deployed on the ground at random chosen locations to be observed by real sensors. Thus, the test is not only of the proper functioning of the probability calculator, but also of the correct modeling of the sensors and of the correspondence of the assumed prior distribution function with the actual distribution of object deployments. By keeping track of various intermediate data streams and distributions the component (sensor, observation model, prior distribution, or probability calculator) of the system responsible for error would be identifiable in the event the system failed to pass the test.

However, each instantiation of ground truth would be costly, and these would have to be repeated many times in order to accrue an adequate sample histogram for comparison with the output of the fusion engine or Tracker. No one would really undertake such a costly test, so the question arises as to what can be accomplished by simulating on a computer the ground deployments and the responses of the sensor to each instantiation of ground truth.

Clearly, a simulation could not test the validity of the prior distribution model or of the sensor observation model. We must either simulate the ground truth instances using the same prior or latent distribution that is assumed by the fusion engine or we build some degree of risk of failure into the test procedure. The same goes for the correspondence between the distribution used to simulate a sample of sensor messages for each ground truth and observation model that is provided to the fusion engine to interpret the messages. Thus, a test using simulated instantiations of ground truth and messages can test only the fusion engine. This is no great disappointment, because it is the fusion engine that we set out to test in the first place.

At the outset of the discussion we established that the fusion engine is a deterministic part of the system and that its verification is easily achieved by simply comparing the output of the fusion engine to the result it should give computed by some independent means. The test using a sample of simulated instances of ground truth and associated messages can be looked upon as just such a test in which a Monte Carlo calculation is used as the means for independently computing the desired distribution. Because a Monte Carlo calculation is not itself a deterministic process, there is some risk that a correctly implemented fusion engine will be judged to have failed the test, but this risk is under our control in the choice of the sample size and can be made as small as we desire, and the Monte Carlo calculation can always be replaced with a deterministic calculation as a final arbiter of a failed test.

A benefit of using the Monte Carlo process for the independent calculation of the output distribution is that it provides an intuitively accessible demonstration of the relationship between the instantiations of ground truth and the distributions of probability that are computed by the fusion engine or Tracker. This relationship is rather less apparent if the calculations done by the fusion engine are checked by more efficient and straightforward means. It is important to make manifest this relationship, because some of the test procedures that have been proposed in the past suggest misunderstanding and unrealistic expectations regarding what can be accomplished by a fusion engine such as Tracker. This problem is not unique to this community. A recent book describing how ideas regarding the quantification of chance have affected the natural and social sciences includes the following:

> . . . No amount of mathematical legerdemain can transform uncertainty into certainty, although much of the appeal of statistical inference techniques stems from just such great expectations. These expectations are fed by ignorance of the existence of alternative theories of statistical inference, by the conflation of calculated solutions with unique ones, by the reduction of objectivity to intersubjective consensus, and above all by the hope of avoiding the oppressive responsibilities that every exercise of personal judgment entails. It would be unjust to blame the mathematical statisticians for these false hopes, although some of their number have shared them. Rather, the fascination with mechanized inference stems from more widespread yearnings for unanimity in times of strife, and for certainty in uncertain circumstances. [Gigerenzer, et al., 1989]

5.4. EXTENSION TO MORE EXTENSIVE SCENARIOS

The above discussions identified a procedure by which a conventional statistic, for example, a chi-squared test, could be used to compare the distribution of belief evoked from Tracker by an observational message, to a distribution of ground truths that could have given rise to that message from an imperfect sensor. Here we wish to consider how that method could be extended to test scenarios more elaborate than a single observation.

The chief problem to be overcome in extending the test procedure from a single observation to more complex scenarios is one of combinatorial complexity. It is assumed that we have available to us a generator of scenarios and a generator of corresponding messages, that are consistent with the behavioral and observational models that have been inputted to Tracker. These generators must be run over and over, creating instantiated pairings of randomly generated scenarios and message streams. Only those instantiated pairings can be used that lead to a message stream matching that which has been selected for the test. Pairings that do not meet this condition are to be discarded. Because so vast is the space of possible combinations of scenarios and message streams, only a minuscule fraction of the pairs generated can be utilized in any given test. It is possible that some form of sequential generation and testing can be employed to make the process more efficient, but it would seem that if carried to the extreme, such a process would reduce the process to one of testing on the basis of one message at a time in the manner discussed above.

6. REFERENCES

Bhattacharyya, A., ``On a measure of divergence between two statistical populations defined by their probability distributions," Bull. Calcutta Math. Soc., vol.35, pp. 99-109, 1943.

DeWitt, R. N., 1997, "Managing Tracking Hypotheses by Combining,'' 1997 IRIS National Symposium on Sensor and Data Fusion.

Gigerenzer, G., Swijtink, Z., Porter, T., Daston, L., Beatty, J., and Kruger, L., 1989, "The Empire of Chance," Cambridge University Press, Cambridge.

Howson, C. and Urbach, P., 1989, "Scientific Reasoning, The Bayesian Approach," Open Court, Chicago.

Lakatos, Imre, 1978, "The Methodology of Scientific Research Programmes," Cambridge University Press, Cambridge.

Popper, Karl R., 1959, "The Logic of Scientific Discovery," Routledge, London.